

Algorithm & Flowcharts

Algorithm:

Step 1:

input w, L

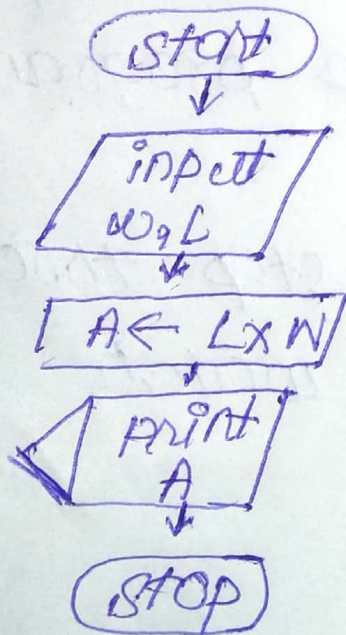
Step 2:-

$A \leftarrow L \times W$

Step 3:-

Print A

Flow chart:



(b) name different type of error which can occur during the execution of a program

→ there are three kinds of errors: syntax errors, runtime errors and logic errors. These are errors where the compiler finds something wrong with your program, and you can't even try to execute it. For example, you may have incorrect punctuation, or may be trying to use a variable that hasn't been declared.

Syntax Errors:

Syntax errors are the easiest to find and correct. The computer will tell you where it got into trouble, and its best guess as to what you did wrong. Usually the error is on the exact line indicated

by the compiler, on the line just before it, however, if the problem is incorrectly nested braces, the actual error may be at the beginning of the nested block.

Runtime Errors:-

If there are no syntax errors Java may detect an error while your program is running. You will get an error message telling you the kind of error, and a stack trace that tells not only where the error occurred, but also what ~~also~~ other method you were in.

Logic:-

It is an error when your program compiles and runs, but does the wrong thing. The Java system of course, has no idea what your program is ~~system~~ supposed

to do so it provides no additional information to help you find the error.

ways to track a logic error include:

- Think about what the program must do in order to produce the results it did. This will lead you to where the error must have occurred.
- Put in print statements to help you figure out what the program is actually doing.
- Use a debugger to step through your program and watch what it does.

Q2 why we use `iostream.h` and `conio.h` in C++.

> `iostream.h`

This is the name of library definition file for all input output streams. Your program will almost certainly want to send stuff to the screen and read things from keyboard.

→ `iostream.h` is the only name of the file in which has code to do that work for you.

`conio.h` :-

`conio.h` is the header file used to include some functions like `clrscr()`, `getch()`, etc. You can include these functions by simply `#include <conio.h>` where `conio` stands

for console input output.

In the C programming language `conio.h` is used for include some predefined function which are already defined in C library.

So we can directly use them.

As `conio.h` we also often use a header file known as `stdio` stands for standard input output.

What do you understand by the term "Maintain and update the program"?

Maintain and update are modification of software product after delivery to correct faults, to improve performance or other attributes, or to adapt the product to a modified environment.

Program maintenance is the process of modifying a software program after delivery to achieve any of these outcomes,

- correct errors
- improve performance
- Add functionalities
- remove obsolete portions.

Despite the common perception that maintenance is required to fix errors that come up after the software goes live, in reality most of the maintenance work involves adding minor or major capabilities to existing modules.

Types of maintenance:

i) corrective maintenance:

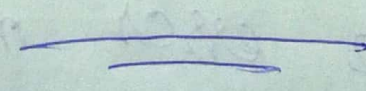
Here errors that come up after on-site implementation are fixed. The error may be pointed out by the users themselves.

preventive maintenance:-
Modification done to avoid
error in future are called
preventive maintenance.

perfective maintenance
changes does in the
existing software to incorporate
new requirements from the client
is called perfective maintenance.

Adaptives
changes in the working
environment sometimes require modifi-
cations in the software. This is
called adaptive maintenance.

For example, if government education
policy changes corresponding changes
have to be made in student result
processing module of school
management software.



Q.1

low level language & High level language
High level language
is easy but low level language
are not easy. High level
language are easy to modify
while low level are not easy
to modify. High level allow much
more abstraction but low level
allow little or no abstraction.
Low is close to machine
language, while high or level is
machine independent programming language.

Q.2

system & application software?
system software is used
for operating computer hardware.
On other hand application
software is used by user
to perform specific task. System
software are installed on the
computer when operating system
is installed. application software
installed according to user requirement.

(1) Compilers & Assemblers-

Compilers translate high level programming language code to machine level code. Assemblers convert the assembly level language to a machine level code. Compilers check and convert the complete code at one time. Assemblers generally does not convert complete code at one time.

(2) Bug & debug:

Bug are errors in code of your program that make your program function improperly. While fixing bugs is called debugging, the process of detecting & removing of existing errors.

(3) Syntax error & logical error:

A system error is an error in the source code of

a program where a logic error is a "bug" or mistake in a program's source code that results in an ~~error~~ or unexpected behaviour.

